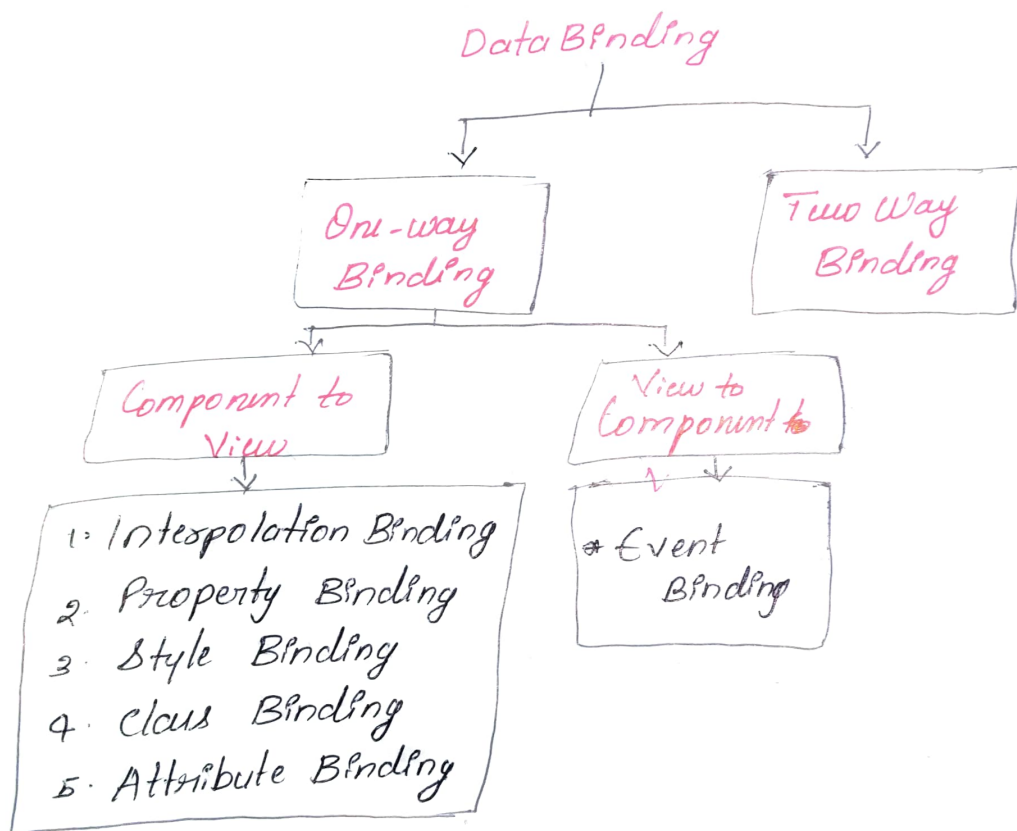


Day  
6-4-20

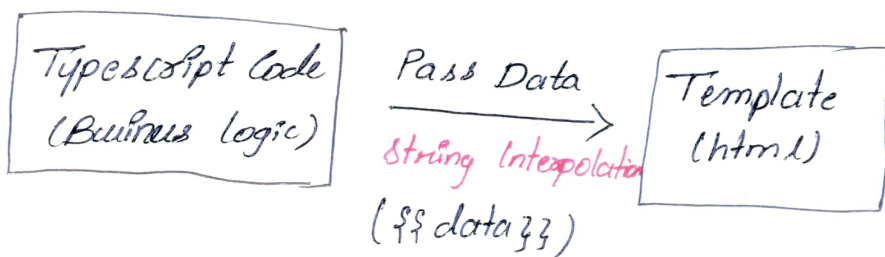
## Data Binding

- Data Binding means to bind the html content with components field (typescript)
- Data binding means interacting with data. In other words we can say it as the interaction between templates (html) and business logic (typescript) it is called binding the data.

### Types of data-binding.



### Interpolation:-



- (15)
- \* Interpolation is a technique that allows the user to bind the data to a User Interface (UI) element.
  - \* Interpolation binds the data in one way

### app.component.ts

```
import { Component } from '@angular/core';  
  
@Component({  
  selector: 'app-root',  
  templateUrl: './app.component.html',  
  styleUrls: ['./app.component.css']  
})  
export class AppComponent  
{  
  title = 'data Binding';  
}
```

### app.component.html

```
<h1>Welcome to angular class {{ title }} </h1>
```

### Output:-

Welcome to angular class data Binding

↓  
string interpolation

### index.html

```
<html>  
<head> </head>  
<body>  
  <app-root> </app-root>  
</body>  
</html>
```

### app.component.html

```
<h1> Our project title  
is {{ title }} </h1>
```

### app.component.ts

```
import { Component } from  
  '@angular/core';  
  
@Component({  
  selector: 'app-root',  
  templateUrl: - - - -  
  styleUrls: - - - -  
})  
export class AppComponent  
{  
  title = 'data Binding'  
}
```

# Creating Variable and displaying it on template

## app.component.ts

```
import { Component } from '@angular/core';  
@Component({  
  selector: 'app-root',  
  templateUrl: './app.component.html',  
  styleUrls: ['./app.component.css']  
})  
export class AppComponent {  
  title = 'data Binding'  
  public name = 'Vinat';  
  jersey: number = 18;  
  wifeName: string = 'Anushka'  
}
```

## app.component.html

```
<h1>We are in appcomponent</h1>  
<p>hello we are in {{name}}</p>  
<p>my jersey number is {{jersey}}</p>  
<p>my wife name is {{wifeName}}</p>
```

## Output:-

We are in app Component

hello we are in Vinat

my jersey number is 18

my wife name is Anushka

} static  
data.

(16)  
Adding function in typescript file.

app.component.ts

```
import { Component } from '@angular/core'
```

```
@Component({
```

```
  selector: 'app-root'
```

```
  templateUrl: './app.component.html'
```

```
  styleUrls: ['./app.component.css'],
```

```
})
```

```
export class AppComponent {
```

```
  title = 'data Building'
```

```
  myFun() {
```

```
    return 'writing function in app.component.ts file'
```

```
  }
```

```
}
```

app.component.html

```
<h1>We are in app component </h1>
```

```
<p>functions: {{ myFun() }} </p>
```



Output:-

We are in app component

function: writing function in app.component.ts file

⇓  
Static Data

whenever we call the function the function body will be activated and get executed.

## String interpolation & advantages

1. It supports expression
2. String interpolation can do string concatenation
3. Every object in javascript are typescript has properties and methods
4. Properties will be without parenthesis. ( ( ) )

### Properties & methods.

#### app.component.ts

```
import { Component } from '@angular/core';  
@Component({  
  selector: 'app-root',  
  templateUrl: './app.component.html',  
  styleUrls: ['./app.component.css']  
})  
export class AppComponent {  
  title = 'dataBinding';  
}
```

#### app.component.html

```
<h1> we are in appcomponent </h1>  
<h1> {{ 10+10 }} </h1>  
<h1> {{ "welcome to " + title }} </h1>  
→ <h1> {{ title.length }} </h1>  
→ <h1> {{ title.toUpperCase() }} </h1>  
→ <h1> {{ title.toLowerCase() }} </h1>
```

#### Output:-

```
We are in appcomponent  
20  
welcome to dataBinding  
11  
DATABINDING  
databinding
```



Limitation with respect to string interpolation

\* Binding cannot contain assignment operator

{{a = 2 + 2}}

\* In string interpolation the boolean values cannot be acceptable.

### Data Binding:-

\* One way binding from ts file to template

\* It is used to bind the values of component properties to the HTML elements




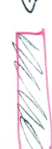



\* Syntax is [property] = expression

example of databinding w.r. to input tag

app.component.html

```
<input id="myId" type="text">
<input id="{{myId}}" type="text">
<input [id]="myId" type="text">
<input disabled="true" id="{{myId}}" type="text">
<input disabled="false" id="{{myId}}" type="text">
<input [disabled]="false" id="{{myId}}" type="text">
<input [disabled]="true" id="{{myId}}" type="text">
```

Inspect

```
<input -ngcontent-lh-a-c0="" id="myId" type="text"> → 
<input "" type="text" id="angular"> → 
<input "" type="text" id="angular"> → 
<input "" disabled="true" type="text" id="angular"> → 
<input "" disabled="false" type="text" id="angular"> → 
<input "" disabled="true" type="text" id="angular"> → 
<input "" type="text" id="angular"> → 
```

## app.component.ts

```
import { Component } from '@angular/core';
```

```
@Component({
```

```
  selector: 'app-root',
```

```
  templateUrl: './app.component.html'
```

```
  styleUrls: ['./app.component.css']
```

```
})
```

```
export class AppComponent {
```

```
  title = 'data Binding'
```

```
  public myId = "angular"
```

```
}
```

Disabled is the html property by default it has boolean true value.